

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-078178

(43)Date of publication of application : 23.03.1999

(51)Int.Cl. B41J 29/38  
B41J 21/00  
G06F 3/12

(21)Application number : 09-262666 (71)Applicant : OKI DATA:KK

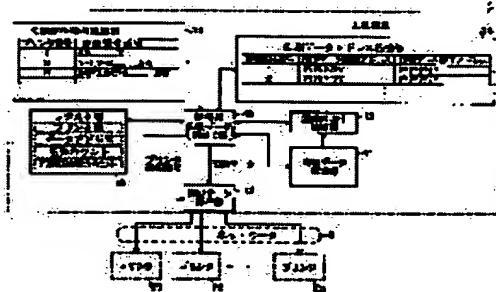
(22)Date of filing : 10.09.1997 (72)Inventor : YOSHIDA TATSUO  
YAMASHITA MAKOTO  
KAI TAKUYA  
ITO KAZUHIKO

## (54) PRINT SYSTEM

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To perform printing quickly and efficiently by an arrangement wherein a host unit connected with a plurality of printers through a network detects the partitioning data of a print medium in a print data, partitions the print data and assigns the print data to a plurality of printers which can request printing.

**SOLUTION:** In a system where a host unit 1 is connected with a large number of printers P (P<sub>1</sub>-P<sub>n</sub>) through a network 3, the print data assigning section, i.e., the control section 10, of the host unit 1 is connected with a print data memory section 11, a medium partition detecting section 12, a print data delivery section 13, a print data address memory section 14, or the like. A print data being delivered to the printer P is stored in the print data memory section 11 and the partition of print medium is detected for the stored print data at a medium partition detecting section 12. Subsequently, each printer P is assigned with a print data based on the number of print media required for printing the entire print data and the number of printers which can request printing and then printing is executed.



**LEGAL STATUS**

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision-of-rejection]-

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2000 Japan Patent Office

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-78178

(43) 公開日 平成11年(1999)3月23日

(51) Int.Cl.<sup>®</sup>  
B 41 J 29/38  
21/00  
G 06 F 3/12

識別記号

F I  
B 41 J 29/38  
21/00  
G 06 F 3/12

Z  
Z  
D

審査請求 未請求 請求項の数10 FD (全 17 頁)

(21) 出願番号 特願平9-262666

(22) 出願日 平成9年(1997)9月10日

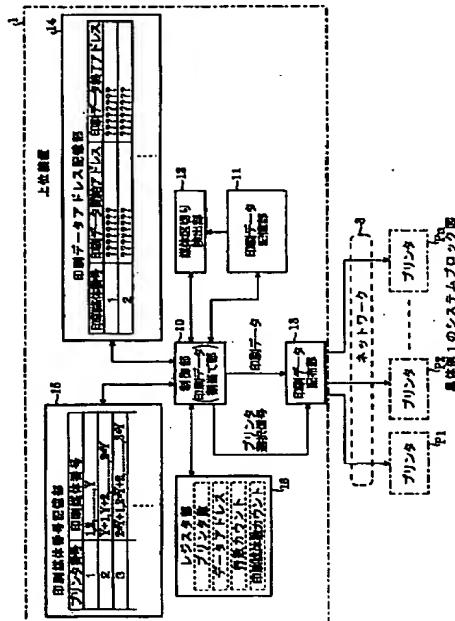
(71) 出願人 591044164  
株式会社沖データ  
東京都港区芝浦四丁目11番地22号  
(72) 発明者 吉田 達穂  
東京都港区芝浦四丁目11番地22号 株式会  
社沖データ内  
(72) 発明者 山下 誠  
東京都港区芝浦四丁目11番地22号 株式会  
社沖データ内  
(72) 発明者 関 琢也  
東京都港区芝浦四丁目11番地22号 株式会  
社沖データ内  
(74) 代理人 弁理士 佐藤 幸男 (外1名)  
最終頁に続く

(54) 【発明の名称】 印刷システム

(57) 【要約】

【解決手段】 上位装置1には印刷データを実際に印刷させた場合の印刷媒体の区切りを検出する媒体区切り検出部12が設けられる。区切り検出後の印刷データは、印刷要求が可能な複数のプリンタP1, P2, … Pnに自動的に分配される。

【効果】 ネットワーク3に接続された印刷要求が可能な複数台のプリンタに、分担して印刷を要求するので、効率よく迅速に印刷結果が得られる。



## 【特許請求の範囲】

【請求項1】 上位装置と複数のプリンタとがネットワークを介して相互に接続されている場合に、

上位装置には、

印刷しようとする印刷データ中の、印刷媒体の区切り情報を検出する媒体区切り検出部と、

前記印刷データ全てを印刷するために必要な印刷媒体数と、印刷要求の可能なプリンタ数とを求めて、印刷要求の可能な複数のプリンタに印刷データを割り当てる印刷データ割当て部と、  
前記割当てに従って前記印刷データを分割し、該当するプリンタに配付する印刷データ配付部とを備えたことを特徴とする印刷システム。

【請求項2】 請求項1に記載のシステムにおいて、印刷データ割当て部は、

印刷データ全てを印刷するために必要な印刷媒体数と印刷要求の可能なプリンタ数に基づいて、印刷媒体数がほぼ等しくなるように、各プリンタに印刷データを割り当てる 것을 特徴とする印刷システム。

【請求項3】 請求項1に記載のシステムにおいて、

印刷データ配付部は、

各プリンタに一定の順番を設定し、先頭のプリンタから順番に、それぞれ1媒体分ずつ印刷データを配付し、最後のプリンタに1媒体分の印刷データを配付したとき、未だ配付していない印刷データがある場合には、再び先頭のプリンタから順番に残りの印刷データを1媒体分ずつ配付するようにして、印刷データ全てを配付し印刷させることを特徴とする印刷システム。

【請求項4】 請求項1に記載のシステムにおいて、印刷データ割当て部は、

印刷媒体ごとの印刷データ量と、各プリンタの印刷速度に基づいて、印刷終了までの時間がほぼ等しくなるように、各プリンタに印刷データを割り当てるることを特徴とする印刷システム。

【請求項5】 請求項1に記載のシステムにおいて、印刷データ割当て部は、

印刷媒体ごとの印刷データの形態と各プリンタの印刷特性に基づいて、各印刷データを印刷するのに最適な特性のプリンタを選択して、該当する印刷データを各プリンタに割り当てるることを特徴とする印刷システム。

【請求項6】 請求項1に記載のシステムにおいて、印刷データ配付部は、

各プリンタに一定の順番を設定し、先頭のプリンタから順番に、空いているプリンタを探索し、最初に検出したプリンタに対して、1媒体分の印刷データを配付するという動作を、全ての印刷データを配付し終わるまで繰り返すことを特徴とする印刷システム。

【請求項7】 請求項1から6のうちのいずれかに記載のシステムにおいて、

印刷データ配付部は、各プリンタに出力すべき印刷要求

を最初に受取り、ネットワークに接続された各プリンタのスプーラの状況を観察するダミードライバにより構成されることを特徴とする印刷システム。

【請求項8】 請求項1から6のうちのいずれかに記載のシステムにおいて、

上位装置には、

印刷要求を出して印刷を実行したプリンタと、その印刷データに関する情報を表示する表示部が設けられていることを特徴とする印刷システム。

【請求項9】 請求項1から6のうちのいずれかに記載のシステムにおいて、

上位装置には、

印刷要求を出すべきプリンタを指定する操作部が設けられていることを特徴とする印刷システム。

【請求項10】 請求項1から6のうちのいずれかに記載のシステムにおいて、

上位装置には、

印刷要求を出す前に、その印刷データに関する情報を表示する表示部と、

印刷要求を出すプリンタを変更して指定する操作部とが設けられていることを特徴とする印刷システム。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、上位装置から複数のプリンタに対して印刷要求を行う印刷システムに関する。

## 【0002】

【従来の技術】パーソナルコンピュータやワードプロセッサ等によって文書を作成すると、その結果をプリンタに出力することが行われる。通常、プリンタはパーソナルコンピュータやワードプロセッサ等に直接接続される。また、ワードプロセッサとプリンタとが一体になつた構成のものもある。この他に、LAN(ローカルエリアネットワーク)等のネットワークに接続された何台かのコンピュータが同様にしてネットワークに接続されたプリンタを共用し、必要な印刷を行うシステムもある。ネットワークに数台のプリンタが接続されていれば、いずれかのプリンタを指定して、そのプリンタに印刷データを送る。

## 【0003】

【発明が解決しようとする課題】ところで、上記のような従来のシステムには次のような解決すべき課題があった。ネットワークに複数台のプリンタが接続され、ネットワークに接続された多数のコンピュータがこれらのプリンタを共用する場合には、それぞれ空いているプリンタを調べ、その空いているプリンタに対して印刷データを送信する。

【0004】ここで、印刷データが大量にあるような場合、印刷データを受信したプリンタは長時間、そのデー

タの印刷を実行する。ところが、この間、他のプリンタは必ずしも有効に利用されているとは限らない。もし、プリンタの使用頻度が低いようであれば、こうした印刷に他のプリンタも同時に活用することが望ましい。しかしながら、オペレータが印刷しようとする文書を適当に切り分けて複数のプリンタに個別に印刷要求をするという処理は非常に煩雑である。しかも、1つの文書を切り分ける場合、その切り分け方法を誤ると、ページの区切りが不自然になり、希望する印刷出力を得られない。

## 【0005】

【課題を解決するための手段】本発明は以上の点を解決するため次の構成を採用する。

〈構成1〉上位装置と複数のプリンタとがネットワークを介して相互に接続されている場合に、上位装置には、印刷しようとする印刷データ中の、印刷媒体の区切り情報を検出する媒体区切り検出部と、上記印刷データ全てを印刷するために必要な印刷媒体数と、印刷要求の可能なプリンタ数とを求めて、印刷要求の可能な複数のプリンタに印刷データを割り当てる印刷データ割当て部と、上記割当てに従って上記印刷データを分割し、該当するプリンタに配付する印刷データ配付部とを備えたことを特徴とする印刷システム。

【0006】〈構成2〉構成1に記載のシステムにおいて、印刷データ割当て部は、印刷データ全てを印刷するために必要な印刷媒体数と印刷要求の可能なプリンタ数に基づいて、印刷媒体数がほぼ等しくなるように、各プリンタに印刷データを割り当てる特徴とする印刷システム。

【0007】〈構成3〉構成1に記載のシステムにおいて、印刷データ配付部は、各プリンタに一定の順番を設定し、先頭のプリンタから順番に、それぞれ1媒体分ずつ印刷データを配付し、最後のプリンタに1媒体分の印刷データを配付したとき、未だ配付していない印刷データがある場合には、再び先頭のプリンタから順番に残りの印刷データを1媒体分ずつ配付するようにして、印刷データ全てを配付し印刷させることを特徴とする印刷システム。

【0008】〈構成4〉構成1に記載のシステムにおいて、印刷データ割当て部は、印刷媒体ごとの印刷データ量と、各プリンタの印刷速度に基づいて、印刷終了までの時間がほぼ等しくなるように、各プリンタに印刷データを割り当てる特徴とする印刷システム。

【0009】〈構成5〉構成1に記載のシステムにおいて、印刷データ割当て部は、印刷媒体ごとの印刷データの形態と各プリンタの印刷特性に基づいて、各印刷データを印刷するのに最適な特性のプリンタを選択して、該当する印刷データを各プリンタに割り当てる特徴とする印刷システム。

【0010】〈構成6〉構成1に記載のシステムにおいて、印刷データ配付部は、各プリンタに一定の順番を設

定し、先頭のプリンタから順番に、空いているプリンタを探索し、最初に検出したプリンタに対して、1媒体分の印刷データを配付するという動作を、全ての印刷データを配付し終わるまで繰り返すことを特徴とする印刷システム。

【0011】〈構成7〉構成1から6のうちのいずれかに記載のシステムにおいて、印刷データ配付部は、各プリンタに出力すべき印刷要求を最初に受取り、ネットワークに接続された各プリンタのスプーラの状況を観察するダミードライバにより構成されることを特徴とする印刷システム。

【0012】〈構成8〉構成1から6のうちのいずれかに記載のシステムにおいて、上位装置には、印刷要求を出して印刷を実行したプリンタと、その印刷データに関する情報を表示する表示部が設けられていることを特徴とする印刷システム。

【0013】〈構成9〉構成1から6のうちのいずれかに記載のシステムにおいて、上位装置には、印刷要求を出すべきプリンタを指定する操作部が設けられていることを特徴とする印刷システム。

【0014】〈構成10〉構成1から6のうちのいずれかに記載のシステムにおいて、上位装置には、印刷要求を出すべきプリンタを選択した後、印刷要求を出す前に、その印刷データに関する情報を表示する表示部と、印刷要求を出すプリンタを変更して指定する操作部とが設けられていることを特徴とする印刷システム。

## 【0015】

【発明の実施の形態】以下、本発明の実施の形態を具体例を用いて説明する。

〈具体例1〉図1は、具体例1の印刷システムを示すブロック図である。図のシステムでは、上位装置1が多数のプリンタP1, P2, …Pnとネットワーク3を介して接続されている。上位装置1は、パーソナルコンピュータやワードプロセッサ等から構成される。なお、このネットワーク3には、図示しない任意の数の上位装置1が接続されていてよい。プリンタP1～Pnの数は2台以上任意である。

【0016】上位装置1には、上位装置1を制御する制御部10が設けられている。この制御部10は、後で説明する印刷データ割当て部として動作する。制御部10には、印刷データ記憶部11、媒体区切り検出部12、印刷データ配付部13、印刷データアドレス記憶部14、印刷媒体番号記憶部15及びレジスタ部16が接続されている。印刷データ記憶部11には、上位装置1が印刷を要求し、プリンタP1, P2, …Pnに配付するための印刷データが格納される。媒体区切り検出部12は、後で説明する手順で、印刷データ記憶部11に記憶された印刷データを解析し、印刷媒体の区切りを検出する部分である。

【0017】印刷媒体とは、プリンタで印刷を行う際に

使用する用紙等をいう。印刷データは、そのデータ量に応じて、2ページ以上の印刷媒体にまたがって印刷されることが多い。この場合、1ページ目から2ページ目の印刷媒体に移る部分を媒体区切りと呼ぶことにする。この区切りを検出することによって、媒体毎の印刷データの範囲が明確になる。この範囲を印刷データアドレス記憶部14に記憶する。即ち、印刷データアドレス記憶部14には、印刷媒体番号と印刷データ開始アドレスと印刷データ終了アドレスとがそれぞれ対応するように格納される。

【0018】印刷媒体番号というのは、印刷媒体毎に順番に付けられた番号で、この番号が“1”という場合、1ページ目の媒体を示し、“2”という場合、2ページ目の媒体を示す。印刷データ開始アドレスは、印刷データ記憶部11中の該当する印刷媒体番号のデータの先頭アドレスを示す。印刷データ終了アドレスは、そのページに印刷を行う最終アドレスを示す。従って、この印刷データアドレス記憶部14に表示されたアドレスの範囲の印刷データを読み出すことによって、印刷媒体毎の印刷データを取り出すことができる。

【0019】印刷媒体番号記憶部15には、プリンタ番号と印刷媒体番号とを対応付けて記憶する。印刷データ割当て部10が、後で説明する要領でプリンタ毎に印刷を要求する印刷媒体番号を指定すると、その結果が、この印刷媒体番号記憶部15に記憶される。ここでは、例えればプリンタ番号が1番のプリンタには、印刷媒体番号が1番～Y番の印刷媒体を印刷させるという結果が示されている。この例では、各プリンタに、それぞれ等しくYページずつ印刷をさせるように制御している。

【0020】レジスタ部16には、こうした印刷データの割当てや印刷データの配付処理を行う際に必要なパラメータが格納されている。このパラメータには、例えばこの図に示すように、プリンタ数、データアドレス、行数カウント、印刷媒体数カウント等がある。印刷データ配付部13は、制御部（印刷データ割当て部）10が印刷媒体番号記憶部15に割り当てた結果に基づいて、各プリンタP1, P2, …Pnに対し印刷要求を行い、必要な印刷データを配付する処理を実行する部分である。なお、実際には、媒体区切り検出部12や印刷データ配付部13あるいは印刷データ割当て部は、いずれも制御部10等の実行するコンピュータのプログラム等により構成される。

【0021】図2には、上記媒体区切り検出部12の動作フローチャートを示す。まず、ステップS1において、印刷データ記憶部11に記憶された印刷データの最初のデータアドレスを印刷データアドレス記憶部14の印刷媒体番号の印刷データ開始アドレスに記憶する。この場合、最初であるから、印刷媒体番号“1”的印刷データ開始アドレスが決定する。次に、ステップS2において、レジスタ部16の印刷媒体数カウントを“1”に

設定し、データアドレスを印刷データの開始アドレスから1引いた値に設定する。こうして初期化処理が完了する。次のステップS3において、行数カウントを“0”に初期化する。そして、ステップS4において、データアドレスの値に“1”を足す。これで最初の印刷媒体用の印刷データからデータの読み出しが可能になる。

【0022】ステップS5では、データアドレスに示されたアドレスの印刷データを1バイト読み込む。更に、ステップS6で、読み込んだ印刷データは改行コードかどうかを判断する。改行コードでなければステップS7に進み、読み込んだ印刷データは改ページコードかどうかを判断する。読み込んだ印刷データが改行コードならば、ステップS8に進み、行数カウントの値に“1”を足す。ステップS9では、行数カウントが1印刷媒体の最大行数かどうかを判断する。即ち、ここで最大行まで印刷を行った場合に媒体の区切りを検出する。また、ステップS7で、改ページコードの有無を判断した場合には、印刷媒体の区切りと判断する。

【0023】ステップS9で行数が印刷媒体の最大行数に満たない場合にはステップS4に戻り、ステップS4からステップS7の処理を繰り返す。ステップS7やステップS9で印刷媒体の区切りを検出した場合には、ステップS10に進み、現在のデータアドレス値を、印刷媒体数カウント値と同じ印刷媒体番号の印刷データ終了アドレスに記憶する。そして、印刷媒体数カウント値に“1”を足す。更に、ステップS11で、現在のデータアドレスに“1”を足した値を印刷媒体数カウントの値と同じ印刷媒体番号の印刷データ開始アドレスに記憶する。即ち、ここで、印刷データのアドレスを“1”だけ進めて、ステップS3に進み、行数の初期化を行って、ステップS4からステップS7の処理を繰り返す。

【0024】一方、ステップS7において、改ページコードでないと判断された場合には、ステップS12に進み、読み込んだ印刷データが印刷終了コードかどうかを判断する。印刷終了コードでなければ再びステップS4に進み、次のデータ読み込みを繰り返す。

【0025】こうして、印刷データ中で媒体区切りを検出すると、その都度図1に示す印刷データアドレス記憶部14に印刷データ開始アドレスや印刷データ終了アドレスを書き込む。そして、全ての印刷データについて処理を終了すると、ステップS13に進み、現在のデータアドレス値を印刷媒体数カウント値と同じ印刷媒体番号の印刷データ終了アドレスに記憶する。これは最後のページの印刷データ終了アドレスを記入する処理である。

【0026】次に、印刷データ割当て部即ち図1に示す制御部10が、印刷データアドレス記憶部14に記憶された内容を参照して、各プリンタに印刷データを割り当てる動作を説明する。図3には、印刷媒体番号記憶部への記憶動作説明図を示す。図1に示したレジスタ部16に記憶したプリンタ数は、ネットワーク3に接続された

プリンタ数である。これは予め所定のタイミングで調査をして記憶する。また、印刷媒体数カウントは、媒体区切り検出部12の処理によって記録される。これらの値を元に1台のプリンタに何ページの印刷媒体の印刷を要求すればよいかを計算する。

【0027】即ち、図3の(1)式に示すように、全体の印刷媒体数をプリンタ数で除し、その結果がY、余りがAという演算処理を行う。割り切れた場合には、各プリンタにYページの印刷を要求する。割り切れない場合には、この図に示すように、例えばプリンタ番号の若い順に1枚ずつ追加分を割り当てる。例えば、この図に示すように、プリンタ数がP1～P4までの4台とし、印刷媒体数が15枚とすると、Yは3、余りが3となる。従って、プリンタP1には4ページ分、プリンタP2、P3にも4ページ分、プリンタP4には3ページ分の印刷媒体について印刷要求を行うことになる。こうした結果を図1に示す印刷媒体番号記憶部15に記憶する。

【0028】次に、印刷データ配付部13の動作を説明する。印刷データ配付部13は、上記のような準備作業が終了すると、プリンタ番号の若い方から順に各プリンタを選択するためのプリンタ選択信号を出し、そのプリンタに対して割り当てられた印刷媒体番号の印刷データを順に配付する。なお、図1に示す印刷媒体番号記憶部15と印刷データアドレス記憶部14の内容を参照すれば、どのプリンタに対しても任意のタイミングで、任意の印刷媒体番号の印刷データを送信することが可能になる。

【0029】図4には、印刷データ配付部の動作フローチャートを示す。まず、ステップS1において、送信終了ページ数を“0”に設定し、パラメータiとjを“1”に初期化する。iは印刷媒体番号記憶部15に記憶された印刷媒体番号の順番を示す。即ち、あるプリンタに印刷媒体番号が1番、2番、3番の3ページの印刷媒体が割り当てられているとき、i=1ならば1番、i=2ならば2番、i=3ならば印刷媒体番号3番の媒体を示すことになる。jはプリンタのプリンタ番号で、若い順に付けられるパラメータである。jが1番なら最初のプリンタP1、jが2番ならば次のプリンタP2を示す。

【0030】ステップS2では、プリンタjに対し i 個の印刷データが割り当てられているかどうかを判断する。即ち、以下の処理ループで、割り当てられた印刷媒体数全ての送信処理を行うから、例えば3ページ分の印刷データが割り当てられている場合、iが“4”ならば、ステップS2からステップS6に進む。一方、iが“3”以下ならステップS3に進む。ステップS3では、プリンタjに対し i 個目に割り当てられた印刷媒体番号の印刷データを送信する。そして、ステップS4において、送信終了ページ数を“1”だけインクリメントする。次に、ステップS5では、送信終了ページ数が全

てのページ数に該当したかどうかを判断する。全てのページの配布が終了すれば、この処理を抜ける。

【0031】ステップS5において、残りのページが存在すると判断すると、ステップS6に進み、jがNかどうかを判断する。Nというのは、プリンタの印刷要求可能な台数である。即ち、既に割当てを行ったプリンタ数がN台ある場合には、jがNになったかどうかを判断し、Nになつていない場合には、ステップS7に進み、jをインクリメントする。また、jがNになつていれば、ステップS8に進み、jを再び“1”に戻し、iをインクリメントする。即ち、こうして、例えばプリンタP1、P2、P3の3台が存在する場合、プリンタP1、P2、P3と1ページ分ずつ印刷データを送信した後、再びプリンタP1に戻り、次のページの印刷データを送信するといった配付動作が行われることになる。

【0032】〈具体例1の効果〉以上のように、印刷データを印刷媒体の区切り毎に分割し、分割した印刷データを印刷要求可能な複数台のプリンタに配付することによって、1台のプリンタに集中的に印刷を依頼するよりも効率よく迅速に印刷を行うことが可能になる。

【0033】〈具体例2〉上記の例では、印刷要求を受け付けるプリンタの能力がそれぞれほぼ等しく、印刷要求を行う印刷媒体数がほぼ等しくなるように印刷データを割り当てる。そして、印刷データの転送と印刷を効率よく行うために、1ページ分ずつプリンタ番号の若いものから順番に印刷データを配付し、先頭のプリンタから最後のプリンタまで印刷データを配付して、まだ配付していない印刷データがある場合には、再び先頭のプリンタから順番に残りの印刷データを配付するようにした。

【0034】ところが、こうした順番で印刷データを配付すると、例えば印刷速度の速いプリンタは早く印刷を終了し、印刷速度の遅いプリンタは次の印刷要求があつてもまだ印刷中であるといった状態になり得る。同じ印刷媒体を同じ能力のプリンタで印刷しても、印刷されるデータ量が違えばやはり印刷速度に差が生じる。具体例2においては、こうしたプリンタの印刷速度を考慮した制御を行う。

【0035】図5には、具体例2の制御情報説明図を示す。具体例2の制御を行う場合にも、図1と同様の構成の上位装置が使用される。なお、その制御情報の内容が異なるため、この図5に制御情報の内容のみを図示した。図の(a)に示すプリンタの印刷速度テーブル17には、プリンタ番号と各プリンタの平均印刷速度とが対応付けて記憶されている。これらの情報は、ネットワークにプリンタが接続された際に各プリンタに問合せを行い、記憶しておく。(b)には、印刷媒体番号記憶部15の内容を示した。これは、具体例1と同様に、プリンタ番号に対応する印刷媒体番号を記憶したものである。この例ではプリンタの印刷速度を考慮した割り当てが行われる。

【0036】(c)には、印刷データ関連情報記憶部の内容を示した。ここには、印刷媒体数カウントレジスタ18A、印刷データアドレス記憶部18B、印刷媒体単位のデータ量記憶エリア18C、全体のデータ量記憶エリア18D、印刷媒体単位のデータ量割合記憶エリア18E、データアドレスレジスタ18F及び行数カウントレジスタ18Gが設けられている。

【0037】この印刷データ情報記憶部18の印刷媒体単位のデータ量記憶エリア18Cには、印刷データアドレス記憶部18Bを参照した結果に基づく印刷媒体毎のデータ量を記憶する。印刷媒体毎に印刷データの量がまちまちであれば、これも印刷速度に影響するためである。また、全体のデータ量記憶エリア18Dには、印刷データ全体のデータ量を記憶する。これによって、各印刷媒体は全体のデータ量に対し何パーセントのデータ量であるかといった比率計算を後で行う。印刷媒体単位のデータ量割合記憶エリア18Eには、この計算結果が記憶される。この他のデータは、具体例1に示したレジスタ部16等に格納したデータやパラメータと同様である。

【0038】図6には、具体例2による印刷データ割当て動作フローチャートを示す。まず、ステップS1において、図5(c)に示した印刷データアドレス記憶部18Bの各印刷媒体毎の印刷データ開始アドレスや印刷データ終了アドレスを元に、各印刷媒体単位のデータ量を算出し記憶する。この記憶先は、印刷媒体単位のデータ量記憶エリア18Cである。

【0039】次に、ステップS2において、各印刷データのデータ量を合計し、全体のデータ量として記憶する。これを図5(c)に示す全体のデータ量記憶エリア18Dに記憶する。ステップS3では、全体のデータ量に対する各印刷媒体単位のデータ量の割合R1、R2、R3…を記憶する。これを図5(c)に示す印刷媒体単位のデータ量割合記憶エリア18Eに記憶する。

【0040】次に、ステップS4において、各プリンタに印刷させる媒体の番号を決定し、その印刷媒体番号を印刷媒体記憶部に記憶する。このとき、図6の左側に示したような演算処理を実行する。即ち、 $Q_n = R_1 + R_2 + R_3 + \dots$ という式によって各プリンタに印刷させる印刷データ量の割合を求める。nはプリンタ番号である。そして、各プリンタの印刷データの割合 $Q_n$ とそのプリンタの印刷速度 $V_n$ との比がほぼ一定になるように要求を行う印刷媒体を割り当てる。

【0041】図7には、印刷データの例説明図を示す。ここには、各印刷媒体毎に印刷データ量MBと全体のデータ量に対する割合Rxが示されている。この全体のデータ量に対する割合Rxを合計すると100になる。また、図8には、印刷速度の例説明図を示す。

【0042】ここには、プリンタ番号に対応して各プリンタの平均印刷速度が示されている。ここで、印刷速度

の比率Vxを求める。印刷速度が速いプリンタほど多くのデータ量を持つ印刷媒体を割り当てれば、ほぼ同時に印刷を開始した場合に、印刷の終了もほぼ同時刻となる。このような観点から図6のステップS4の左下に示したように、印刷速度に応じた印刷媒体が割り当てられる。この例では、プリンタ番号が1番のプリンタに対して印刷媒体番号が2番の印刷媒体を割り当てている。プリンタ番号が1番のプリンタの印刷速度の比率がV1である。また、プリンタ番号が2番の印刷媒体のデータ量の割合がR2である。

【0043】同様にして、プリンタ番号が2番のプリンタに印刷媒体番号が3番の印刷媒体を割り当て、プリンタ番号が3番のプリンタに印刷媒体番号が1番の印刷媒体を割り当てる。また、プリンタ番号が4番のプリンタには印刷媒体番号が4番と5番の印刷媒体を割り当てる。これらの比がいずれもほぼ等しく“1”に近い値になれば、最も効率よく各プリンタに印刷データを割り当てることになる。その後の制御は具体例1の場合と同様である。

【0044】〈具体例2の効果〉複数のプリンタがそれぞれ印刷速度が異なり、また、それぞれ別々の内容のデータ量の印刷媒体を印刷する場合に、それぞれ同程度の時間で印刷を終了させることができる。なお、印刷データ量を考慮せず、印刷速度の比のみによって印刷媒体の割当を行いうようにしても差し支えない。

【0045】〈具体例3〉印刷データには、グラフィックデータのみを印刷するページ、テキストデータとグラフィックデータの入り混じった内容を印刷するページ、テキストデータだけのページ等が存在する。こうした場合、同一のプリンタであっても印刷速度が異なってくる。また、プリンタの中には、解像度が高く、グラフィックデータの印刷に適したものやテキストデータを高速で印刷することができるものがある。本発明においては、こうした印刷データの内容を印刷データの形態と表現し、また、プリンタの性能等を印刷特性と表現し、この具体例において各印刷データを印刷するのに最適な特性のプリンタを選択する制御を行う。

【0046】図9には、具体例3の制御情報説明図を示す。具体例3の印刷も図1に示したような概略構成の上位装置が使用される。この上位装置の制御情報として、具体例3では、ここに示すような内容のものが利用される。

【0047】図の(a)には、印刷形態記憶部19の内容を示した。ここには、印刷媒体番号に対応させてその印刷データの内容が示されている。なお、この印刷データの内容は、例えば600dpiのテキストデータ(Text)とグラフィックデータ(Gra)、300dpiのテキストデータとグラフィック、用紙の大きさがA4判とA3判というように分類されている。

【0048】そして、例えば印刷媒体番号1番の印刷デ

ータは、この図に示すように600dpiのグラフィックデータ部分にフラグ1が立てられている。即ち、この印刷媒体には、600dpiのグラフィックデータのみが印刷される。また、2番の印刷媒体番号の印刷データは300dpiのテキストデータとグラフィックデータが印刷される。なお、1番の印刷媒体番号については、使用される用紙がA3判、2番の印刷媒体番号については、使用される用紙がA4判となっている。こうした印刷形態に応じたプリンタの選択が行われることになる。

【0049】(b)には、プリンタの印刷特性記憶部20の例を示した。ここには、プリンタ番号に対応して各プリンタの印刷特性が示されている。この印刷特性は、印刷形態記憶部19の内容と対応させ、600dpiのテキストデータとグラフィックデータ、300dpiのテキストデータとグラフィックデータ、印刷可能な用紙の判数等が表示されている。例えば、プリンタ番号1番については、600dpiのテキストデータとグラフィックデータの印刷に適し、また用紙はA4判、A3判のいずれかを使用することができる。2番のプリンタ番号のプリンタについては、600dpiのグラフィックデータと300dpiのテキストデータやグラフィックデータが印刷可能で、用紙はA4判のみが利用できるといった内容になっている。

【0050】図10には、媒体区切り検出部の動作フローチャートを示す。この媒体区切り検出部が、具体例1で説明したような動作と共に印刷データの印刷形態を判断し、図9の(a)に示したようなデータを生成する。この動作は、図2に示した媒体区切り検出部の動作と同様の手順で、図1に示した印刷データ記憶部11からデータを読み出し(ステップS4)、その中に含まれる印刷形態コードを検出する(ステップS5)。そして、各印刷媒体毎に、どのような内容のどのような形態の印刷データが含まれるかを検出し、図9(a)に示すような印刷形態記憶部に記憶する動作を行う。従って、ステップS5とステップS6は、この部分固有の動作を示すが、その他の部分は図2に示したようなアドレスを更新しながらデータを順に読み取る処理と同一である。

【0051】即ち、現在読み込みを行っている印刷媒体の読み込んだ印刷データは、印刷形態コードかどうかをその都度判断する(ステップS5)。そして、印刷形態コードでなければその判断処理を終える。また、印刷形態コードであれば、ステップS6に進み、印刷形態記憶部の印刷形態データの該当するビットを“1”にする。即ち、その印刷データが300dpiのテキストデータを示しているものであれば、該当する部分のビットを“1”にすることによって、印刷形態記憶部への記録を終了する。こうした処理を全ての印刷データについて印刷媒体毎に実行することによって、図9に示す印刷形態記憶部へのデータが完成する。

【0052】図11に、印刷データ配付部の動作フロー

チャートを示す。この処理は、図4を用いて示した具体例1の印刷データ配付部の動作と類似している。ところが、具体例1では、各プリンタに最初の印刷媒体から順に1印刷媒体分ずつ印刷データを画一的に転送していた。この図11に示す例では、印刷媒体の印刷形態と、プリンタの特異な印刷形態即ちそのプリンタの印刷特性とが一致しているかどうかを判断する。そして、一致している場合に、そのプリンタに該当する印刷媒体を割り当てる。この処理がステップS2とステップS3である。その他の処理は図4を用いて説明した処理と変わることはない。

【0053】〈具体例3の効果〉以上により、印刷データの内容が様々な形態に分かれていた場合に、最もその印刷に適したプリンタに印刷データを送信して印刷を行わせることができる。従って、印刷データを分担させて複数のプリンタに印刷させるだけでなく、より高画質な印刷出力を得ることもできる。なお、印刷データが全てテキストデータであって、全てのプリンタがテキストデータの印刷を行うことができるものであれば、具体例1のような印刷データの配付を行えばよい。また、印刷データが全てグラフィックであって、特定のプリンタのみしか印刷ができないような場合には、そのプリンタにのみ印刷要求を行えばよい。

【0054】〈具体例4〉具体例1においては、例えば4台のプリンタが存在する場合に、第1ページ目は1番目のプリンタP1に、第2ページ目は2番目のプリンタP2に、第3ページ目は3番目のプリンタP3に、第4ページ目は4番目のプリンタP4に、第5ページ目は1番目のプリンタP1に、というように自動的に順番に印刷媒体の割当てを行った。従って、印刷要求可能なプリンタが4台あれば4台のプリンタに均等に分散して印刷が要求される。

【0055】ところが、プリンタに順番に1ページ分ずつ印刷データを転送しながら印刷要求を行うと、何番目かのプリンタに印刷要求を行っている間に最初に印刷要求を行ったプリンタの印刷が既に終了していることがある。こうした場合に、プリンタ番号順に並んだ次のプリンタに印刷要求を行ってもよいが、既に印刷を終了したプリンタに印刷を要求することもできる。

【0056】ここで、印刷終了後の媒体を回収する作業を考えると、印刷要求を行うプリンタ数は少ないほど印刷媒体の回収は容易である。これを考慮すれば、次々に新たなプリンタに印刷要求を行うよりも、過去に印刷要求を行っており、既に印刷動作が終了したプリンタに印刷要求を改めて行う方がよいこともある。この具体例はそうした場合の制御を示している。

【0057】図12には、具体例4の動作説明図を示す。例えば、この図に示すように、プリンタP1, P2, P3, P4の4台のプリンタがある場合に、1ページ目をプリンタP1、2ページ目をプリンタP2、3ペ

ページ目をプリンタP3に印刷要求したとする。3ページ目の印刷要求を行い、次の印刷要求を行うタイミングでは、既にプリンタP1は印刷を終了しているとする。この場合、プリンタP4に4ページ目の印刷を依頼せず、プリンタP1に4ページ目の印刷を依頼する。これによって、実際に利用するプリンタが4台から3台に減少し、印刷媒体の回収が容易になる。

【0058】図13には、具体例4と他の例との動作比較説明図を示す。図の(a)は具体例1の印刷要求の手順を示し、プリンタP1, P2, P3, P4まで印刷を依頼すると、再びプリンタP1に戻って、P1, P2, P3という依頼を行う。(b)が、具体例4の例で、最初にプリンタP1に1ページ目の印刷を要求すると、次の2ページ目の印刷要求の際、再び、プリンタP1が空いていないかどうかを判断し、次にプリンタP2が空いていないかどうかを判断するというように、常に先頭から順に空いているプリンタを探す。そして、空いていることが検出された最も番号の若いプリンタに印刷を依頼する。これによって、番号の若いプリンタ側に集中して印刷を依頼することが可能になる。

【0059】図14には、この具体例4の動作フローチャートを示す。まず、ステップS1において、印刷データアドレスを読み込む。この場合の印刷データアドレスは具体例1の印刷データアドレス記憶部14に格納されたとの同様の内容でよい。そして、印刷データを印刷媒体1枚分ずつ取り出して、次のステップに進む。

【0060】ステップS2では、印刷されていないページがあるかどうかを判断する。即ち、既に準備された印刷データの全てが処理されていればステップS8にジャンプする。印刷要求すべきページがあればステップS3に進み、送信ページを決定する。例えば先頭の印刷媒体から順番に処理する場合にパラメータを用意し、現在パラメータによって指定されている印刷媒体のページが決定される。

【0061】ここで、ステップS4において、プリンタ状態の読み込みを行う。即ち、印刷要求を行うべきプリンタからはそのプリンタが印刷を既に終了して次の印刷要求を受け付けられる状態にあるかどうかの情報を受け入れる。そして、ステップS5において、印刷可能なプリンタがあるかどうかを判断する。なお、このステップS4とステップS5で、図13(b)に示したような手順で空いているプリンタの探索を行う。そして、ステップS6において、該当のページのプリンタへの割当てを決定し、ステップS7において、1ページ分の印刷データを送信する。これで、選択したプリンタへの1ページ分の印刷データ送信が終了し、再びステップS8からステップS1に戻って、次の印刷媒体についての印刷データ転送処理に進む。

【0062】〈具体例4の効果〉以上のように、プリンタにプリンタ番号等を付けて、一定の順番を設定し、常

にその先頭からプリンタの状態を探索し、最初に空き状態と判断したプリンタに対して印刷媒体の印刷を要求するようすれば、印刷を要求するプリンタが印刷番号の若いものに集中するため、後から印刷媒体の回収をすることが容易になる。

【0063】〈具体例5〉上記具体例1～具体例4のような動作を実行させるために、例えば上位装置がパソコンコンピュータの場合、各パソコンコンピュータには、それぞれ印刷要求を行うプリンタを制御するためのプリンタドライバが必要となる。しかしながら、各パソコンコンピュータ上のアプリケーションプログラムは、通常、1つのプリンタドライバに対する印刷要求を行う機能のみを備えている。

【0064】従って、この具体例では、アプリケーションプログラムが印刷要求を行うダミードライバを設定し、このダミードライバが印刷要求を分散させる複数のプリンタ用のプリンタドライバに関する情報を持ち、必要に応じて印刷データを分配するという構成をとった。これによって、アプリケーションはダミードライバに対し印刷要求を行い、ダミードライバはネットワーク上にある複数のプリンタスプールの状況を観察し、所定の演算処理によって該当するプリンタに対し印刷要求を行うよう動作する。

【0065】図15には、具体例5の構成説明図を示す。この図に示すように、アプリケーション22が印刷要求を行うために、ダミードライバ23が設けられ、このダミードライバ23から印刷要求を受けるプリンタAドライバ25、プリンタBドライバ26及びプリンタCドライバ27が設けられている。ダミードライバ23には、この図に示すような情報24が保持されている。即ち、各プリンタをA, B, Cとすると、各プリンタ毎にポートの種類、ポートに唯一のプリンタ、処理能力優先順位、デフォルトプリンタ等の情報が記憶される。

【0066】例えば、プリンタAはローカルプリンタで、出力ポートが1ポートである。また、プリンタ処理能力は4ppm(枚/分)、印刷解像度は30dpiである。プリンタBは、例えばパソコンコンピュータに接続されたローカルプリンタで、リモート接続されたパソコンコンピュータによって共有されている。そのプリンタ処理能力は8ppm、印刷解像度は600dpiとする。また、プリンタCはTCP/IPネットワークプリンタで、出力ポートはIPアドレス、処理能力は8PPM、印刷解像度が600dpiである。こうした情報がダミードライバ23に格納され、ダミードライバ23はこのような情報を元に印刷データの配付を行う。

【0067】図16には、具体例5の動作フローチャートを示す。この図に示すように、上記ダミードライバが動作する。まず、ステップS1において、アプリケーション22からの印刷要求があると、ステップS2において、各プリンタのスプーラの状態を観測する。観測は、

各スプーラにスプールされているデータがあるかどうか、いくつあるか、データを印刷していない残りのページが何ページ分なのかという情報を返してもらう。この観測結果によって、印刷データをそれぞれプリンタへ送った場合の印刷開始時間を算出する。算出方法は、スプールされているデータのページ数を処理能力で割ることにより算出する。

【0068】次のステップS4では、印刷させるプリンタの選定を行う。プリンタの選定は、印刷開始時間が短いプリンタを選定する。どのプリンタのスプーラにもデータがない場合には、デフォルトの設定のままにし、算出結果が同じ場合には、ダミードライバの持つ優先順位データから優先順位の高いプリンタを設定する。優先順位はこのような目的のために設定された。プリンタ選定後は、該当するプリンタドライバによりデータ作成スプール印刷という制御が行われる。

【0069】〈具体例5の効果〉ネットワーク上にある複数のプリンタスプーラの状況をダミードライバにより観察し、スタッカされている印刷情報の数が少ないプリンタを検出し、そのドライバを自動的にダミードライバに選択させ、印刷要求を行うようにすれば、ネットワーク上にあるプリンタの使用効率を上げて、アプリケーションから印刷要求を出してからプリンタで印刷を終了するまでの時間を短縮することができる。また、自己の印刷ジョブよりも先に送られた他人の印刷ジョブの印刷によって、特定のプリンタに対する自己の印刷ジョブの印刷が思いがけず時間がかかり、最終的に全ての印刷結果を得るのが遅くなるといった問題を解決できる。

【0070】〈具体例6〉上位装置から複数のプリンタに対して印刷要求を行い、自動的に複数のプリンタに分担させて印刷を行う場合には、どのプリンタにどのページを印刷させたかという結果をオペレータに表示する必要がある。また、例えば印刷データの割当て処理を行う前に、オペレータが意図的にいすれかのプリンタを分配の対象から外したり、印刷の優先順位を変更したりすることを可能にしたい。

【0071】図17には、こうした処理を可能とする表示部と操作部の例説明図を示す。この図は、上位装置のディスプレイ等に表示されたウインドウの内容を示している。図に示す表示部31には、例えばネットワークに接続された全てのプリンタのプリンタ名が、例えばここではP1, P2, P3, P4というように表示されている。そして、これらのプリンタに対し、それぞれ印刷要求を行うかどうかを示すチェックボックス33が付加されている。

【0072】また、各プリンタには、何枚印刷要求を行ったかを表示するテキストボックス34が付加されている。これらのチェックボックス33をチェックすることによって、印刷要求を禁止するプリンタを指定したり、図の下側に示すように印刷速度を優先するか、印刷する

画質を優先するかの選択を行うようとする。こうしたチェックボックス33等の制御イメージをここでは操作部32と呼んでいる。もちろん、マウスやキーボード、その他各種のインターフェースから操作の意思を入力するようにして差し支えない。

【0073】図17の印刷制御ウインドウの右端には、印刷依頼をした各プリンタの状態を示すコンボボックス35を図示した。このコンボボックス35には、各プリンタP1, P2, P3, P4が現在印刷を実行する予定のジョブ(JOB)が、実行順にリストされている。印刷を依頼するプリンタを指定すると、自分の依頼に基づくジョブ番号がこのコンボボックス35中に書き込まれる。各コンボボックス35の下向きの矢印をクリックすると、アルダウンリスト中にその内容が表示される。自分の依頼したジョブはリストの最後に表示される。

【0074】この内容を見ると、各プリンタの処理待ちのジョブ数が分かるから、各プリンタ毎に、いつごろ印刷が完了し、いつごろその出力を取りにいかなければ判断することができる。なお、印刷を依頼する前にこのコンボボックスが表示されると、早く印刷結果を得ようとする場合に、依頼先の最適化ができる。

【0075】〈具体例6の効果〉以上のような表示を行うことによって、具体例1～具体例4に示した各種の制御を行い、更に必要に応じてオペレータがその制御の方向性を選択することが可能になる。例えば、4台のプリンタ全てが印刷要求受付可能であっても3台のプリンタにのみ印刷要求を分配するといった制御を行うことができる。また、印刷速度よりも画質を優先し、特性の合ったプリンタに対し該当するデータを送る制御を設定することができる。

#### 【図面の簡単な説明】

【図1】具体例1のシステムを示すブロック図である。

【図2】媒体区切り検出部の動作フローチャートである。

【図3】印刷媒体番号記憶部への記憶動作説明図である。

【図4】印刷データ配付部の動作フローチャートである。

【図5】具体例2の制御情報説明図である。

【図6】具体例2による印刷データ割当て動作フローチャートである。

【図7】印刷データの例説明図である。

【図8】印刷速度の例説明図である。

【図9】具体例3の制御情報説明図である。

【図10】印刷データ割当て部の動作フローチャートである。

【図11】印刷データ配付部の動作フローチャートである。

【図12】具体例4の動作説明図である。

【図13】具体例4と他の例との動作比較説明図であ

る。

【図14】具体例4の動作フローチャートである。

【図15】具体例5の構成説明図である。

【図16】具体例5の動作フローチャートである。

【図17】表示部と操作部の例説明図である。

【符号の説明】

1 上位装置

3 ネットワーク

10 制御部(印刷データ割当て部)

11 印刷データ記憶部

12 媒体区切り検出部

13 印刷データ配付部

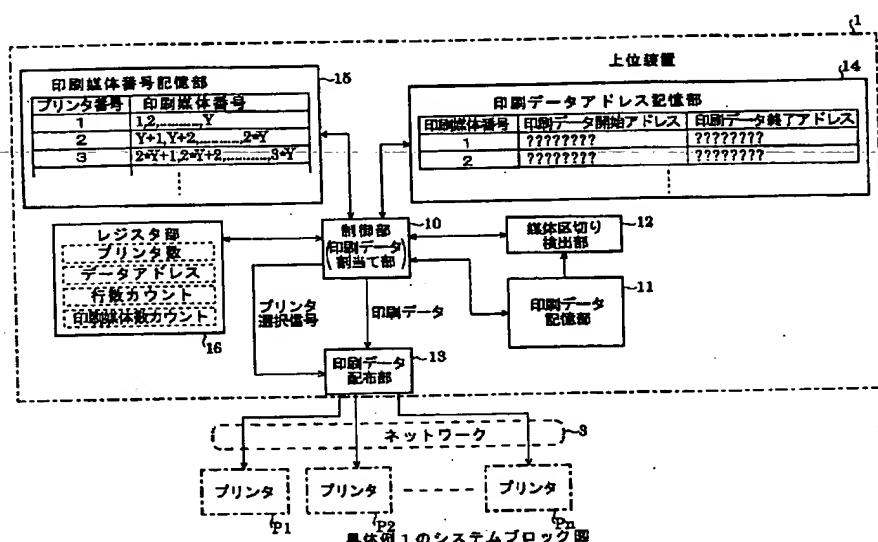
14 印刷データアドレス記憶部

15 印刷媒体番号記憶部

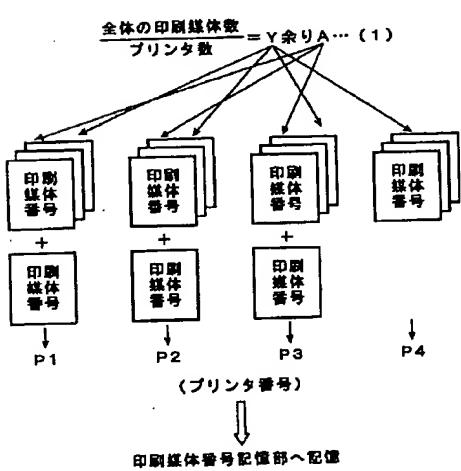
16 レジスタ部

P<sub>1</sub>, P<sub>2</sub>, … P<sub>n</sub> プリンタ部

【図1】



【図3】



【図7】

印刷媒体番号	印刷データ量 (MB)	全体のデータ量に対する割合 R × (%)
1	3	30
2	1	10
3	2	20
4	3	30
5	1	10

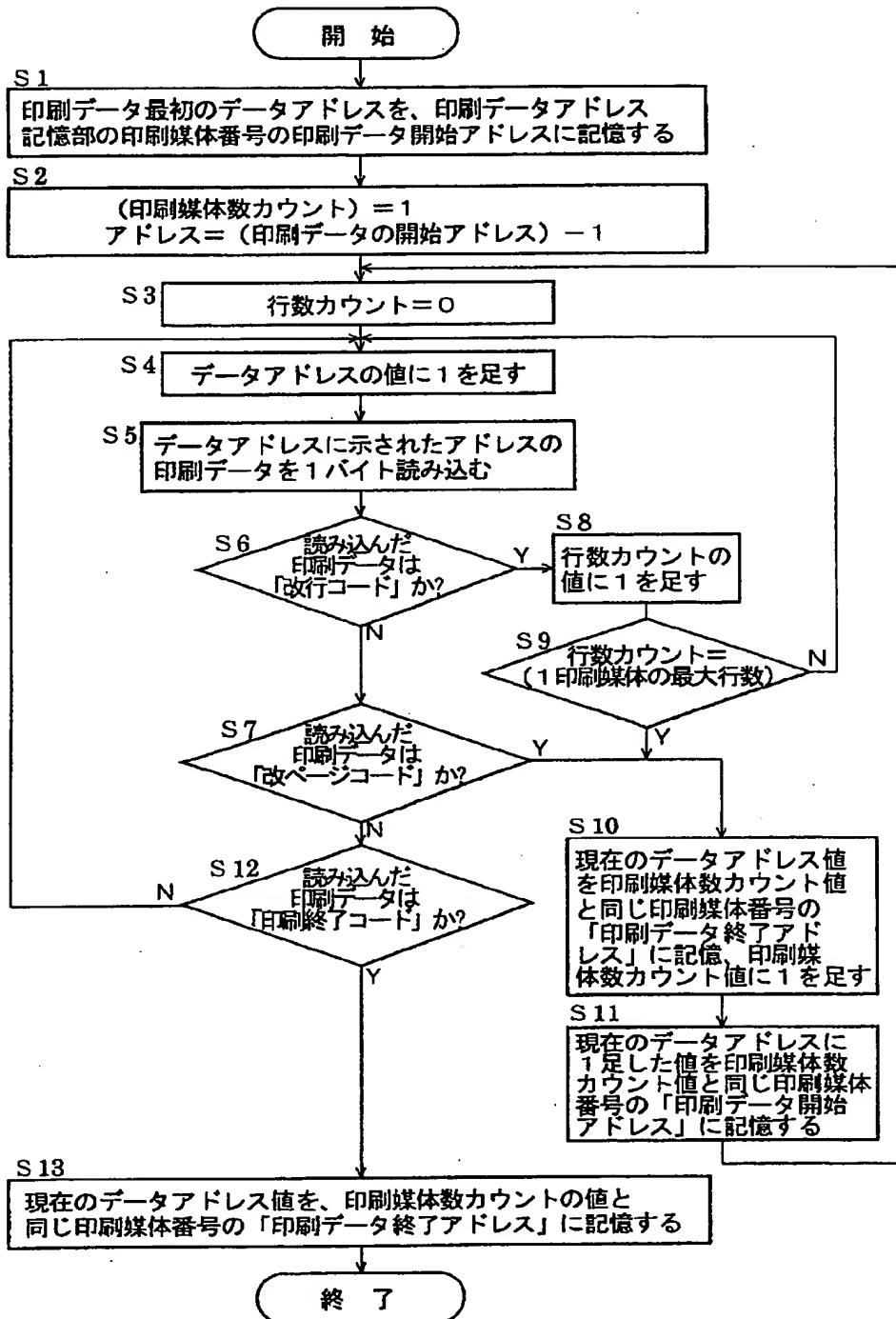
印刷データの例説明図

【図8】

プリンタ番号	平均印刷速度 (ppm)	印刷速度の比率 V ×
1	10	1
2	20	2
3	30	3
4	40	4

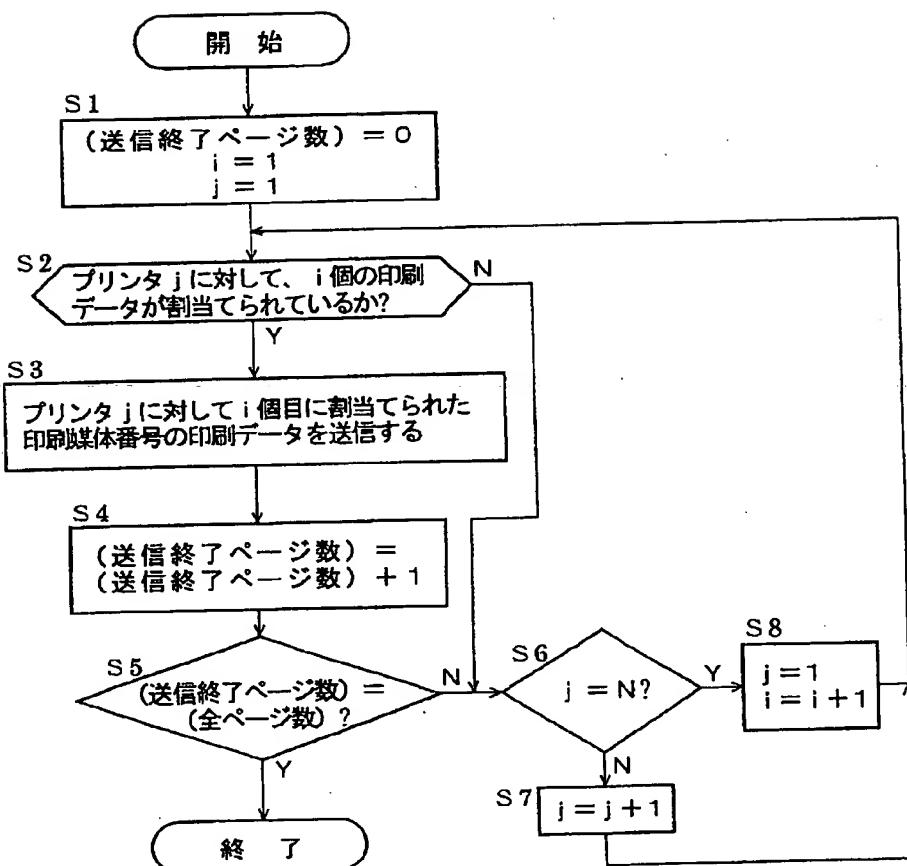
印刷速度の例説明図

【図2】



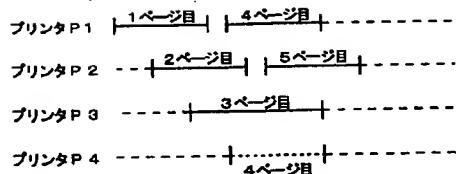
媒体区切り検出部の動作フローチャート

【図4】



印刷データ配布部の動作フローチャート

【図12】



具体例4の動作説明図

【図5】

17 プリンタの印刷速度テーブル	
プリンタ番号:平均印刷速度	
2	xxxxxx yyyyyy
N	xxxxxx

(a)

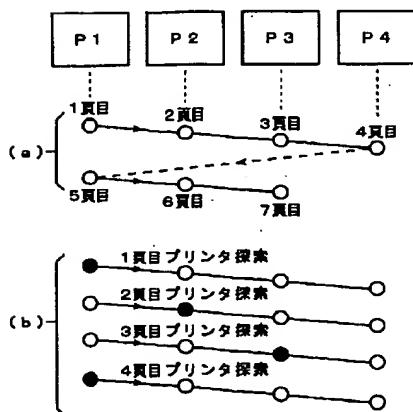
15 印刷媒体番号記憶部	
プリンタ番号:印刷媒体番号	
1	xx, xx, xx,
2	xx, xx, xx,
N	

(b)

印刷データ関連情報記憶部		
印刷媒体数カウントレジスタ		
印刷媒体番号	印刷データ開始アドレス	印刷データ終了アドレス
1	????????	????????
2	????????	????????
印刷媒体単位のデータ量記憶エリア		
全体のデータ量記憶エリア		
印刷媒体単位のデータ量割合記憶エリア		
データアドレスリスト		
行数カウントレジスタ		

具体例2の制御情報説明図

【図13】



具体例4と他の例との動作比較説明図

【図9】

19 印刷形態記憶部	
印刷媒体番号	600dpi
Text	Gra
	300dpi
Text	Gra
	用紙
A4	A3
1	0
2	0
⋮	⋮
M	0

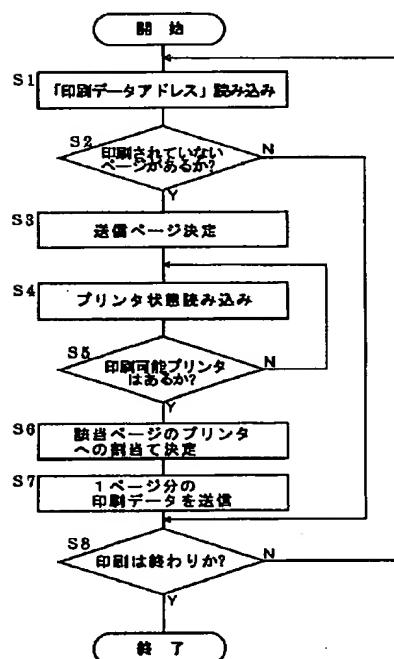
(a)

20 プリンタの印刷特性記憶部	
プリンタ番号	600dpi
Text	Gra
	300dpi
Text	Gra
	用紙
A4	A3
1	1
2	0
⋮	⋮
N	0

(b)

具体例3の制御情報説明図

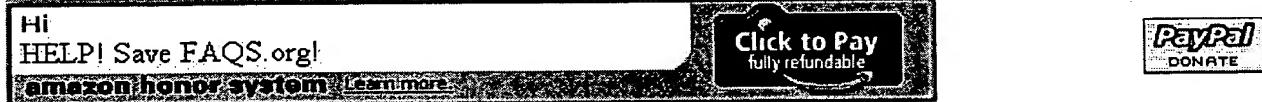
【図14】



具体例4の動作

[Part1](#) - [Part2](#) - [Part3](#) - [Part4](#) - [Part5](#) - [Single Page](#)Top Document: [comp.os.msdos.programmer FAQ part 4/5](#)Previous Document: [- How can I change the switch character to - from /?](#)Next Document: [- Why does my interrupt function behave strangely?](#)

## - How can I write a TSR (terminate-stay-resident utility)?



There are books, and there's code to download.

First, the books:

- \* Ray Duncan's {Advanced MS-DOS}, ISBN 1-55615-157-8, gives a brief checklist intended for experienced programmers. The ISBN is for the second edition, through DOS 4; but check to see whether the DOS 6 version is available yet.
- \* {DOS 5: A Developer's Guide} by Al Williams, ISBN 1-55851-177-6, goes into a little more detail, 90 pages worth!
- \* Pascal programmers might look at {The Ultimate DOS Programmer's Manual} by John Mueller and Wallace Wang, ISBN 0-8306-3534-3, for an extended example in mixed Pascal and assembler.
- \* For a pure assembler treatment, check Steven Holzner's {Advanced Assembly Language}, ISBN 0-13-663014-6. He has a book with the same title out from Brady Press, but it's about half as long as this one.

Next, the code. Some of it is companion code to published articles, which are also listed below:

- \* John English has created a wonderful C++ class for creating TSRs which includes several examples. This one is my personal favorite. It is downloadable as:  
[<ftp://ftp.simtel.net/pub/simtelnet/msdos/cpluspls/tsr100je.zip>](ftp://ftp.simtel.net/pub/simtelnet/msdos/cpluspls/tsr100je.zip)
- \* The Alternate Multiplex Interrupt Specification, downloadable as:  
[<ftp://ftp.simtel.net/pub/simtelnet/msdos/info/altmpx35.zip>](ftp://ftp.simtel.net/pub/simtelnet/msdos/info/altmpx35.zip)  
[<ftp://garbo.uwasa.fi/pc/programming/altmpx35.zip>](ftp://garbo.uwasa.fi/pc/programming/altmpx35.zip)
- \* Ralf Brown's assembly-language implementation of the spec, with utilities in C, is downloadable as:  
[<ftp://ftp.simtel.net/pub/simtelnet/msdos/asmutil/amis1092.zip>](ftp://ftp.simtel.net/pub/simtelnet/msdos/asmutil/amis1092.zip)  
[<ftp://garbo.uwasa.fi/pc/c-lang/amis1092.zip>](ftp://garbo.uwasa.fi/pc/c-lang/amis1092.zip)
- \* Douglas Boling's MASM template for a TSR is downloadable as:  
[<ftp://ftp.simtel.net/pub/simtelnet/msdos/asmutil/template.zip>](ftp://ftp.simtel.net/pub/simtelnet/msdos/asmutil/template.zip)
- \* Code for Al Stevens's "Writing Terminate-and-Stay-Resident Programs", Computer Language, February 1988, pages 37-48 and March 1988, pages 67-76 is downloadable as:  
[<ftp://ftp.simtel.net/pub/simtelnet/msdos/c/tsrc.zip>](ftp://ftp.simtel.net/pub/simtelnet/msdos/c/tsrc.zip)

Finally, there are commercial products, of which Tessera (for C-language TSRs) is one of the best known.

---

Top Document: [comp.os.msdos.programmer FAQ part 4/5](#)

Previous Document: [- How can I change the switch character to - from /?](#)

Next Document: [- Why does my interrupt function behave strangely?](#)

[Part1](#) - [Part2](#) - [Part3](#) - [Part4](#) - [Part5](#) - [Single Page](#)

---

[ [HELP SAVE FAQS.org!!!!](#) ]

[ [By Archive-name](#) | [By Author](#) | [By Category](#) | [By Newsgroup](#) ]

[ [Home](#) | [Latest Updates](#) | [Archive Stats](#) | [Search](#) | [Usenet References](#) | [Help](#) ]

---

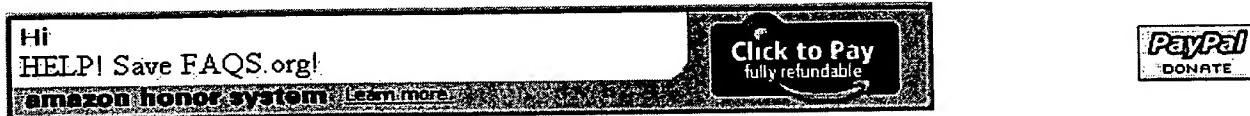
*Send corrections/additions to the FAQ Maintainer:  
[jeffrey@carlyle.com](mailto:jeffrey@carlyle.com) (Jeffrey Carlyle)*

Last Update January 20 2002 @ 11:34 PM

[Part1](#) - [Part2](#) - [Part3](#) - [Part4](#) - [Part5](#) - [Single Page](#)

Top Document: [comp.os.msdos.programmer FAQ part 4/5](#)  
 Previous Document: [- How can I change the switch character to - from /?](#)  
 Next Document: [- Why does my interrupt function behave strangely?](#)

## - How can I write a TSR (terminate-stay-resident utility)?



[We've lost our funding--read more about it.](#)

There are books, and there's code to download.

First, the books:

- \* Ray Duncan's {Advanced MS-DOS}, ISBN 1-55615-157-8, gives a brief checklist intended for experienced programmers. The ISBN is for the second edition, through DOS 4; but check to see whether the DOS 6 version is available yet.
- \* {DOS 5: A Developer's Guide} by Al Williams, ISBN 1-55851-177-6, goes into a little more detail, 90 pages worth!
- \* Pascal programmers might look at {The Ultimate DOS Programmer's Manual} by John Mueller and Wallace Wang, ISBN 0-8306-3534-3, for an extended example in mixed Pascal and assembler.
- \* For a pure assembler treatment, check Steven Holzner's {Advanced Assembly Language}, ISBN 0-13-663014-6. He has a book with the same title out from Brady Press, but it's about half as long as this one.

Next, the code. Some of it is companion code to published articles, which are also listed below:

- \* John English has created a wonderful C++ class for creating TSRs which includes several examples. This one is my personal favorite. It is downloadable as:  
<ftp://ftp.simtel.net/pub/simtelnet/msdos/cpluspls/tsr100je.zip>
- \* The Alternate Multiplex Interrupt Specification, downloadable as:  
<ftp://ftp.simtel.net/pub/simtelnet/msdos/info/altmpx35.zip>  
<ftp://garbo.uwasa.fi/pc/programming/altmpx35.zip>
- \* Ralf Brown's assembly-language implementation of the spec, with utilities in C, is downloadable as:  
<ftp://ftp.simtel.net/pub/simtelnet/msdos/asmutil/amis1092.zip>  
<ftp://garbo.uwasa.fi/pc/c-lang/amis1092.zip>
- \* Douglas Boling's MASM template for a TSR is downloadable as:  
<ftp://ftp.simtel.net/pub/simtelnet/msdos/asmutil/template.zip>
- \* Code for Al Stevens's "Writing Terminate-and-Stay-Resident Programs", Computer Language, February 1988, pages 37-48 and March 1988, pages 67-76 is downloadable as:  
<ftp://ftp.simtel.net/pub/simtelnet/msdos/c/tsrc.zip>

Finally, there are commercial products, of which TessEact (for C-language TSRs) is one of the best known.

---

Top Document: [comp.os.msdos.programmer FAQ part 4/5](#)

Previous Document: [- How can I change the switch character to - from /?](#)

Next Document: [- Why does my interrupt function behave strangely?](#)

[Part1](#) - [Part2](#) - [Part3](#) - [Part4](#) - [Part5](#) - [Single Page](#)

---

[ [HELP SAVE FAQS.org!!!!](#) ]

[ [By Archive-name](#) | [By Author](#) | [By Category](#) | [By Newsgroup](#) ]

[ [Home](#) | [Latest Updates](#) | [Archive Stats](#) | [Search](#) | [Usenet References](#) | [Help](#) ]

---

*Send corrections/additions to the FAQ Maintainer:*

[jeffrey@carlyle.com](mailto:jeffrey@carlyle.com) (*Jeffrey Carlyle*)

Last Update January 20 2002 @ 11:34 PM

**THIS PAGE BLANK (USPTO)**